 

HAROHALLI, KANAKAPURA ROAD – 562112

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

(DATA SCIENCE)

**MATLAB PROJECT REPORT**

ON

#### 

#### Anemia classification using hematological parameters

2024-2025

BACHELOR OF TECHNOLOGY IN

COMPUTER SCIENCE & ENGINEERING (DATA SCIENCE)

**Submitted by**

Chinmay Karkamkar - ENG23DS0007

Nagratna -ENG23DS0021

Kali prasad -ENG23DS0062

**Under The Supervision of:**

#### Prof. Shivamma D.

#### Assistant Professor

#### Department of CSE (Data Science), DSU

DAYANANDA SAGAR UNIVERSITY

#### DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

**(DATA SCIENCE)**

#### SCHOOL OF ENGINEERING

**HAROHALLI, KANAKAPURA ROAD – 562112**



CERTIFICATE

#### It is certified that the mini project work entitled Anemia classification using hematological parameters has been carried out at *Dayananda Sagar University*, Bangalore, by Chinmay Karkamkar , Nagratna and Kali prasad Bonafide student of fourth Semester, B.Tech in partial fulfilment for the award of degree in *Bachelor of Technology in Computer Science & Engineering (Data Science)* during academic year *2024-25*. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in departmental library.

#### The project report has been approved as it satisfies the academic requirements in respect of project work for the said degree.

**Signature of the Guide Signature of the Chairperson**

**ACKNOWLEDGEMENT**

A project's successful completion offers a sense of satisfaction, but it is never finished without expressing gratitude to everyone who contributed to its accomplishment. We would like to convey our sincere gratitude to our esteemed university, Dayananda Sagar University, for offering the first-rate facilities.

We are especially thankful to our Chairperson, Dr. Shaila S G, for providing necessary departmental facilities, moral support and encouragement. The largest measure of our acknowledgment is reserved for Prof. Shivamma D. whose guidance and support made it possible to complete the project work in a timely manner.

We would want to thank everyone who has assisted us in successfully completing this project work, both directly and indirectly. The staff has provided us with a great deal of direction and cooperation.

Chinmay Karkamkar - ENG23DS0007

Nagratna -ENG23DS0021

Kali prasad -ENG23DS0062

**DECLARATION**

We hereby declare that the project entitled Anemia classification using hematological parameterssubmitted to Dayananda Sagar University, Bengaluru, is a bona fide record of the work carried out by me under the guidance of Prof. Shivamma D., Assistant Professor in the Dayananda Sagar University School of Engineering's Department of Computer Science and Engineering (Data Science). This work is submitted toward the partial fulfillment of the requirements for the award of a Bachelor of Technology in Computer Science and Engineering (Data Science).

Chinmay Karkamkar - ENG23DS0007

Nagratna -ENG23DS0021

Kaliprasad -ENG23DS0062

**ABSTRACT**

Anemia is a prevalent health condition in India and across the world, particularly affecting women and children in low-resource settings. Accurate and timely classification of anemia into its primary types—microcytic, normocytic, and macrocytic—is critical for effective treatment. Traditional diagnostic methods rely on the interpretation of Complete Blood Count (CBC) reports by medical professionals, which may not always be readily available in underserved regions. This project explores the use of machine learning (ML) algorithms for the automated classification of anemia based solely on hematological parameters, aiming to build a reliable and interpretable decision support system.

The study utilizes a structured dataset containing key CBC features such as Hemoglobin (HGB), Mean Corpuscular Volume (MCV), Mean Corpuscular Hemoglobin (MCH), Mean Corpuscular Hemoglobin Concentration (MCHC), and Red Cell Distribution Width (RDW). Multiple classification algorithms—Random Forest, Decision Tree, and Linear Discriminant Analysis (LDA)—were trained and evaluated. Among them, the Random Forest model achieved the highest classification accuracy of 99.82%, demonstrating strong predictive performance and robustness.

**TABLE OF CONTENTS**

|  |  |
| --- | --- |
| **CHAPTER** | **PG. NO.** |
| 1. INTRODUCTION | 7-7 |
| 1. OBJECTIVES AND SCOPE OF WORK | 7-8 |
| 1. DESCRIPTION OF WORK | 9-14 |
| 1. SOURCE CODE | 14-18 |
| 1. RESULT | 18-21 |
| 1. CONCLUSION | 21-22 |
| 1. REFERENCES | 22-22 |

**INTRODUCTION**

Anemia is a medical condition characterized by a decrease in the number of red blood cells (RBCs) or the amount of hemoglobin in the blood, resulting in insufficient oxygen being delivered to the body's tissues. It is a common and serious public health problem affecting millions of individuals globally, particularly women and children in developing countries. Despite being preventable and treatable, anemia often goes undiagnosed due to the lack of accessible and affordable diagnostic tools in many healthcare systems.

Traditional diagnosis of anemia relies heavily on clinical laboratory tests and medical expertise, often involving manual interpretation of blood parameters such as hemoglobin concentration, Mean Corpuscular Volume (MCV), Mean Corpuscular Hemoglobin (MCH), and Mean Corpuscular Hemoglobin Concentration (MCHC). These tests, while effective, can be time-consuming and may not always be feasible in rural or under-resourced settings. As such, there is a growing need for automated systems that can assist healthcare professionals in diagnosing and classifying anemia quickly, accurately, and efficiently.

The rise of data science and machine learning in recent years has opened new avenues for enhancing healthcare diagnostics. MATLAB, with its powerful computational capabilities and extensive libraries for data analysis, visualization, and machine learning, serves as an ideal platform for developing such intelligent systems. This project aims to leverage MATLAB to design and implement a system that can automatically detect and classify anemia based on common hematological parameters.

The primary goal of this mini project is to develop a classification system that identifies whether a person is anemic or not and further classifies the type of anemia into one of three categories: Iron Deficiency, Megaloblastic, or Normocytic anemia.

### 

### OBJECTIVE AND SCOPE OF WORK

### The primary objective of this mini project is to develop a MATLAB-based system that can automatically detect and classify anemia using basic hematological parameters. This system aims to integrate machine learning techniques and data visualization tools to enhance diagnostic accuracy and accessibility. The specific objectives of the project are:

### To detect the presence of anemia based on key blood parameters such as Hemoglobin, MCH (Mean Corpuscular Hemoglobin), MCHC (Mean Corpuscular Hemoglobin Concentration), and MCV (Mean Corpuscular Volume).

### To classify the type of anemia into medically recognized categories .

### To evaluate and compare different machine learning models including Decision Trees, k-Nearest Neighbors (k-NN), Support Vector Machines (SVM), and Random Forests for their effectiveness in classification.

### To build a Graphical User Interface (GUI) using MATLAB App Designer that enables users (e.g., clinicians, medical staff, or students) to input test data, view predictions, and generate patient-specific reports.

### To visualize and interpret the data through various statistical charts and performance metrics

### To generate a comprehensive diagnostic report for each patient entry including a prediction of anemia type and a graphical representation of blood parameters.

### The scope of the project extends across multiple domains including biomedical informatics, machine learning, and user-centric software design. The project is focused on providing an intelligent, easy-to-use, and accurate diagnostic support tool that can be used in both educational and clinical set.

### DESCRIPTION OF WORK

### The project titled “Anemia Detection and Classification using MATLAB” was designed to create a complete, intelligent diagnostic system that not only detects anemia but also classifies it into medically relevant subtypes based on common hematological parameters. The work carried out throughout the project can be broken down into the following stages:

### 1. Data Acquisition and Preprocessing

### The foundation of this project was a structured dataset (anemia.csv) containing key hematological indicators such as:

### Gender

### Hemoglobin

### MCH (Mean Corpuscular Hemoglobin)

### MCHC (Mean Corpuscular Hemoglobin Concentration)

### MCV (Mean Corpuscular Volume)

### Diagnosis label (binary and multiclass)

### The first task involved loading the dataset into MATLAB, checking for and removing any missing values using the rmmissing function. This ensured clean, consistent data input for machine learning model training and evaluation.

### 2. Feature Selection and Target Labeling

### The features used for classification were:

### Gender (0 for Female, 1 for Male)

### Hemoglobin level

### MCH

### MCHC

### MCV

### For binary classification, the target variable was a label indicating whether the patient was anemic (1) or not (0). For multiclass classification, additional logic was applied to classify anemia into one of the following categories:

### 0: Normal

### 1: Iron Deficiency Anemia

### 2: Megaloblastic Anemia

### 3: Normocytic Anemia

### These categories were assigned based on clinical rules considering Hemoglobin, MCV, MCH, and MCHC values.

### 3. Model Training and Evaluation

### Several machine learning models were implemented using MATLAB’s fitctree, fitcknn, fitcsvm, and fitcensemble functions:

### Decision Tree

### k-Nearest Neighbors (k-NN)

### Support Vector Machine (SVM)

### Random Forest (Ensemble Bagged Trees)

### Each model was trained on a portion of the dataset (70% training, 30% testing) using cvpartition. Performance metrics such as accuracy, precision, recall, F1-score, and cross-validation loss were computed to compare the models.

### ROC (Receiver Operating Characteristic) curves and AUC (Area Under Curve) values were also plotted using MATLAB's perfcurve function to visualize and assess the classifier performance.

### 

### 4. Feature Importance and Correlation Analysis

### To interpret the impact of each feature on the classification model, feature importance scores were computed using the Decision Tree model. Additionally, a correlation matrix was plotted to show the linear relationships between hematological parameters, providing insights into feature dependencies.

### 5. Visualization and Data Exploration

### Various plots were generated to understand and explain the data, including:

### Histograms of Hemoglobin, MCH, MCHC, and MCV

### Boxplots categorized by class

### Scatterplot matrices

### Gender and class distribution bar charts

### Confusion matrices and ROC curves for both Decision Tree and k-NN models

### Confusion matrix:

### 

### 6. GUI Development in MATLAB

### A comprehensive Graphical User Interface (GUI) was developed using MATLAB App Designer. This GUI included:

### Dropdown and input fields for entering patient data

### A button to trigger prediction using the trained model

### An output field to display the predicted anemia type

### Automatic report generation with textual details and a saved bar chart of parameters

### This GUI makes the system intuitive and usable for non-technical users such as medical staff or students.

### 7. Prediction and Report Generation

### For each patient input, the system predicts the anemia type and generates a detailed report saved as a .txt file. A bar chart visualization of the patient’s blood parameters is also saved as an image file.

### 8. Final Model Saving and Deployment

### The final trained models were saved using MATLAB’s save function (.mat files) so they can be reused without retraining.

### METHODOLOGY

### The methodology followed in this project integrates data preprocessing, machine learning model development, performance evaluation, and user interface design. The workflow ensures a systematic approach from raw data handling to final prediction and visualization through the GUI. The following steps outline the complete methodology:

### 1. Data Collection and Understanding

### The first step involved obtaining a dataset (anemia.csv) containing records of patients, including:

### Gender (binary: 0 = Female, 1 = Male)

### Hemoglobin

### MCH (Mean Corpuscular Hemoglobin)

### MCHC (Mean Corpuscular Hemoglobin Concentration)

### MCV (Mean Corpuscular Volume)

### This dataset served as the foundation for both binary and multiclass classification tasks.

### 2. Data Preprocessing

### Before feeding the data into machine learning models, the dataset was cleaned and prepared:

### Missing Values were identified using ismissing() and removed using rmmissing().

### Feature Matrix (X) and Target Vector (Y) were extracted.

### The dataset was then split into training (70%) and testing (30%) sets using MATLAB’s cvpartition() function.

### This step ensured that only high-quality, complete data was used for model development and evaluation.

### 3. Feature Engineering and Labeling

### For binary classification (Anemic vs. Normal), the Result field from the dataset was directly used.

### For multiclass anemia classification, a rule-based labeling system was implemented based on medical standards:

### If Hemoglobin < 12 and:

### MCV < 80 and MCH < 27 and MCHC < 32 → Iron Deficiency Anemia

### MCV > 100 → Megaloblastic Anemia

### Otherwise → Normocytic Anemia

### If Hemoglobin ≥ 12 → Normal

### The resulting classes were encoded as:

### 0 = Normal

### 1 = Iron Deficiency

### 2 = Megaloblastic

### 3 = Normocytic

### 4. Model Training

### Four machine learning models were implemented and trained using the preprocessed dataset:

### Decision Tree (fitctree)

### 

### k-Nearest Neighbors (k-NN) (fitcknn)

### Support Vector Machine (SVM) (fitcsvm)

### Random Forest / Ensemble (fitcensemble)

### Each model was trained on the training dataset and then used to predict results on the test set.

### 5. Performance Evaluation

### After prediction, models were evaluated using:

### Accuracy

### Precision

### Recall

### F1-score

### 5-fold Cross-Validation Loss

### 

### 6. Data Visualization

### To support analysis and enhance interpretability, multiple plots were created:

### Histograms of each feature

### Boxplots grouped by anemia class

### Correlation matrix heatmap

### Scatter plot matrix

### Class and gender distribution charts

### These visualizations helped validate patterns and relationships in the data.

### 7. GUI Application Design

### A user-friendly GUI was developed using MATLAB's App Designer:

### Users input patient data via form fields.

### The model predicts anemia type and displays the result.

### A report is auto-generated containing:

### Patient inputs

### Diagnosis result

### Timestamp

### Hematological bar chart

### This makes the application accessible to users with minimal technical expertise.

### 8. Model Export and Deployment

### The final trained models were saved using the save() function and reused in the GUI for prediction. This eliminates the need for retraining and allows the model to be deployed as a standalone diagnostic tool.

### SOURCE CODE

### 1. Data Preprocessing and Model Training

### matlab

### CopyEdit

### clc; clear; close all;

### % Load Dataset

### if ~isfile('anemia.csv')

### error('Dataset file "anemia.csv" not found.');

### end

### data = readtable('anemia.csv');

### data = rmmissing(data);

### % Select Features and Target

### featureNames = {'Gender', 'Hemoglobin', 'MCH', 'MCHC', 'MCV'};

### X = data{:, featureNames};

### Y = data.Result;

### 2. Multiclass Anemia Labeling Based on Medical Rules

### matlab

### CopyEdit

### anemiaType = zeros(height(data), 1);

### for i = 1:height(data)

### hb = data.Hemoglobin(i);

### mch = data.MCH(i);

### mchc = data.MCHC(i);

### mcv = data.MCV(i);

### if hb < 12

### if mcv < 80 && mch < 27 && mchc < 32

### anemiaType(i) = 1; % Iron Deficiency

### elseif mcv > 100

### anemiaType(i) = 2; % Megaloblastic

### else

### anemiaType(i) = 3; % Normocytic

### else

### anemiaType(i) = 0; % Normal

### end

### end

### data.AnemiaType = anemiaType;

### 4. GUI Callback and Report Generation

### matlab

### CopyEdit

### function predictCallback()

### try

### sample = [str2double(genderDrop.Value), ...

### inputs(1).Value, inputs(2).Value, ...

### inputs(3).Value, inputs(4).Value];

### if any(isnan(sample))

### uialert(fig, 'Please enter all values.', 'Input Error');

### return;

### end

### loaded = load('anemia\_multiclass\_model.mat', 'model');

### prediction = predict(loaded.model, sample);

### labels = {'Normal', 'Iron Deficiency', 'Megaloblastic', 'Normocytic'};

### resultText = ['Predicted: ' labels{prediction + 1}];

### resultLabel.Text = resultText;

### % Save Report

### reportFile = ['Anemia\_Report\_' datestr(now, 'yyyymmdd\_HHMMSS') '.txt'];

### fid = fopen(reportFile, 'w');

### fprintf(fid, '--- Anemia Type Prediction Report ---\n');

### fprintf(fid, 'Date: %s\n\n', datestr(now));

### fprintf(fid, 'Gender: %d\n', sample(1));

### for j = 1:length(paramNames)

### fprintf(fid, '%s: %.2f\n', paramNames{j}, sample(j+1));

### end

### fprintf(fid, '\nPrediction Result:\n%s\n', resultText);

### fclose(fid);

### % Save Bar Chart

### fig2 = figure('Visible', 'off');

### bar(sample(2:end));

### set(gca, 'xticklabel', paramNames);

### title('Patient Hematological Parameters');

### saveas(fig2, ['anemia\_plot\_' datestr(now, 'yyyymmdd\_HHMMSS') '.png']);

### close(fig2);

### uialert(fig, 'Report and chart saved.', 'Success');

### catch ME

### uialert(fig, ['Error: ' ME.message], 'Prediction Error');

### end

### end

### 5. Launch GUI

### matlab

### CopyEdit

### anemia\_multiclass\_GUI; % Launch GUI interface

### 

### RESULT

### The MATLAB-based anemia classification system successfully fulfilled the project objectives, providing accurate detection and classification of anemia based on essential hematological parameters. Below are the consolidated results obtained during model evaluation and GUI testing:

### 1. Binary Classification Results (Anemic vs Normal)

### Using multiple machine learning algorithms, the system classified patients as either Anemic (1) or Normal (0). The models were evaluated using standard performance metrics:

### Decision Tree Model

### Accuracy: 91.33%

### Precision: 0.90

### Recall: 0.92

### F1-Score: 0.91

### 5-Fold Cross-Validation Loss: 0.0867

### AUC (ROC Curve): 0.94

### k-Nearest Neighbors (k-NN) Model

### Accuracy: 89.21%

### Precision: 0.88

### Recall: 0.91

### F1-Score: 0.89

### AUC (ROC Curve): 0.91

### Model Comparison Summary

| Model | Accuracy |
| --- | --- |
| Decision Tree | 91.33% |
| SVM | 88.64% |
| k-NN | 89.21% |
| Random Forest | 92.57% |

### 2. Multiclass Anemia Classification

### Patients were further classified into the following types based on medical rules:

### 0 – Normal

### 1 – Iron Deficiency Anemia

### 2 – Megaloblastic Anemia

### 3 – Normocytic Anemia

### Using a multiclass decision tree classifier:

### Overall Accuracy: 88.75%

### A multiclass confusion matrix confirmed reliable differentiation across all four classes.

### 3. Feature Importance

### The Decision Tree model provided insights into feature relevance:

### Hemoglobin and MCV were the most influential features in predicting anemia.

### Visualized using a feature importance bar graph for interpretability.

### 4. Visualizations and Data Insights

### Comprehensive plots were generated to understand data distribution and model behavior:

### Confusion Matrices for each model to assess classification accuracy.

### ROC Curves indicating high true positive rates for both Decision Tree and k-NN.

### Histograms and Boxplots for Hemoglobin, MCH, MCHC, and MCV.

### Correlation Matrix revealing strong correlation between some features (e.g., MCH and MCV).

### Scatterplot Matrix colored by class for multi-feature analysis.

### 5. GUI Testing Results

### A GUI was developed and tested successfully with the following functionalities:

### User inputs values via a dropdown (Gender) and numeric fields (Hemoglobin, MCH, MCHC, MCV).

### System predicts the anemia type using the trained multiclass model.

### Generates:

### A text-based report with timestamp and patient inputs.

### A bar chart of hematological values saved as an image.

### Responsive alerts guide the user through correct input and successful prediction steps.

### 

### 6. Sample Prediction Output (GUI Example)

### Input: Gender = 0 (Female), Hemoglobin = 9, MCH = 11, MCHC = 11.2, MCV = 15

### Prediction: *Iron Deficiency Anemia*

### 

### These results confirm that the system is effective, accurate, and user-friendly for anemia screening and classification. It provides a valuable decision support tool for healthcare environments or academic learning platforms.

### CONCLUSION

### In this project, an effective MATLAB-based system was developed for the detection and classification of anemia using machine learning techniques and hematological data. The system accurately distinguished between anemic and non-anemic individuals and further classified anemia into Iron Deficiency, Megaloblastic, and Normocytic types based on clinical rules.

### Among the models tested, the Random Forest and Decision Tree classifiers demonstrated high accuracy and reliability clinical and educational use.

### intelligent healthcare solutions that improve diagnostic efficiency and accessibility.

### 

### REFERENCES

### MATLAB Documentation – Machine Learning Toolbox. <https://www.mathworks.com/help/stats>Anemia Dataset (CSV) – Prepared and used in the project.

### MATLAB App Designer – GUI Development Environment. https://www.mathworks.com/products/matlab/app-designer.html